Desalination and Water Treatment
www.deswater.com

doi: 10.5004/dwt.2019.24216

163 (2019) 415–420
September

Multi-kernel SVM based multi-feature fusion algorithm for analyzing wastewater membrane permeation in environmental pollution

Aidong Fang, Lin Cui*, Zhiwei Zhang

School of Information Engineering, Suzhou University, Suzhou 234000, China, email: asfad5006@163.com (L. Cui)

Received 24 September 2018; Accepted 1 January 2019

ABSTRACT

Wastewater membrane permeation analysis in environmental pollution is of great significance because water pollution information contains rich semantics. Traditional algorithms based on one or several features of environmental pollution related factors are common ways of wastewater membrane permeation but with unsatisfied performance and disadvantages in complex environmental pollution. Algorithms based on deep machine learning seems better in analyzing wastewater membrane permeation with the cost of a great deal of calculation and training samples but shows unsatisfactory and disadvantaged at the case of small sample field. In this paper, a novel algorithm using multiple feature fusion of geometric features and high-level ones of environmental pollution factors based on support vector machine (SVM) with multi-kernel functions is proposed. By using multiple feature fusion, pollution factors can be detected more effectively in complex scenes. Multi-kernel functions may avoid the limitations of single kernel function, which can improve the performance of the algorithm. Known as a statistical learning theory, SVM is suitable for small sample space classification. Experimental results showed that, the proposed algorithm performs well in small sample field and also has good generalization ability.

Keywords: Environment pollution; Feature fusion; Membrane permeation; Water pollution; Support vector machine

1. Introduction

Text in images usually contains rich semantemes so that scene text detection and recognition is of great significance, which makes it being widely studied and applied in the field of pattern recognition such as vehicle license plate location and recognition, text and video detection based on content, automatic recognition system of notes record, web page filtering system, and so on. Text detection is the key problem of text recognition with different difficulties in various scenes. In general, text detection in documents is easier than the videos and game scenes for reasons of less influence of the illumination variance and noises of background. Text detection in natural scenes is the most difficult because of the factors such as noises of background, illumination variance, title angles of objects, and shadowed part of objects and so on so that there is no satisfied way of text detection in natural scenes. This proposed algorithm focuses on text detection in videos and game scenes [1].

There are two common kinds of algorithms of text detection: algorithms based on geometric features and those based on machine learning. The former detects text with the geometric features of characters but shows weak robustness while the later does by a classifier trained with huge number of samples which shows stronger robustness but much more workload of calculations than that of the former. Algorithm proposed in this paper extracts the geometric features and high-level features such as features of extremal regions (ER) and histogram of oriented gradient (HOG), trains the classifier with the feature vectors composed of the

* Corresponding author.

1944-3994/1944-3986 © 2019 Desalination Publications. All rights reserved.

extracted features above. In consideration of limited learning samples, support vector machine (SVM) classifier is applied in text detection. Experimental results showed the proposed algorithm performs well in small sample field and also has good generalization ability.

2. Extraction of the character features

General ways to set up feature vectors follow two principles: sigma completeness and independence. Sigma completeness ensures the features can describe the characteristics of different types of the research objects while independence means the features are independent of each other.

But in fact, it is very difficult to find out and define satisfied features to set up such feature vectors. In this paper, we commend the geometry parameters, ER parameters and the HOG parameters of text region as text features. Each feature can be regarded as a description of text region from a certain point of view so that the features can be fused [2] to remedy the defects of single feature.

As noted previously, because of the uncertainty of optimum features, features have different performances in text detection, which means it is inappropriate to combine these empirical features into a feature vector. Methods of text detection based on simple fusion of features are not only complicated but also unsatisfactory. In this paper, we first extract geometric features and high-level ones of text regions and then give them different weights so that the method may perform well.

2.1. Geometric features of text regions

It is approved that the character of the stroke width is the inherent property of the text which does not depend on particular language. Stroke width is the most common feature, which means most text regions can be detected with it. At first, images are transformed using stroke width transform [3], pixels with stroke width of less than three are connected into a connected domain according to the consistency of stroke width. In view of the fact that the stroke width of background is not as good as that of text regions, the ratio of the maximum and minimum values of stroke width is taken as criterion of the text regions in order to avoid the possible approximation between text regions and background. In this paper, areas that W_{max}/W_{min} is in range of 0.5 to 2.0 are treated as text candidate regions. Further aspect ratio (w_c/h_c) of text candidate regions is used to filter out non-text regions. In this paper, the aspect ratio ranges from 0.3 to 3, which may be invalid for some specific characters such as "*i*" and Chinese character one that will be detected as text regions by the latter two features given in the algorithm as below.

2.2. ER and characteristics of ER

Neumann delivered the concept of ER [2]: pixels values of outer boundary are strictly greater than those inside the regions.

$$\forall p \in R, q \in \partial R : C(q) > \theta \geq C(p) \tag{1}$$

where θ is the threshold of ER, ∂R is the outer boundary of the region grouped with pixels which borders on the region (R) but does not belong to R.

$$\partial R = \{p \in D \mid R : \exists q \in R : pAq\} \tag{2}$$

pAq means that p is adjacent to q.

ER (r) in condition of the threshold (θ) means the intersection which consists pixels of ER and those with values of θ in condition of the threshold.

$$r = \left(\bigcup u \in R_{\theta-1}\right) \cup \left(\bigcup p \in D : C(p) = \theta\right) \tag{3}$$

After comparing with RGB space and HIS space with an intensity gradient (∇), Neumann pointed out the intensity gradient was approximated by the maximum value of intensity difference of adjacent pixels in I space.

$$C_{\nabla}(p) = \max_{q \in D : pAq} \left\{ \left| C_I(p) - C_I(q) \right| \right\} \tag{4}$$

In order to get a fast classification of ER, descriptor of the region is calculated as shown below.

$$\varnothing(r) = \left(\oplus \varnothing(u)\right) \oplus \left(\varnothing(p)\right) \tag{5}$$

$\varnothing(r)$ isthe descriptor of r, $\varnothing(p)$ is the initialization function of calculating the descriptor of the given pixel (p). The descriptors of ER will be calculated with the increase of the threshold (θ) ranged from 0 to 255.

There are three main features of text ER: length of ER, Euler number of ER and number of horizontal crossing points of ER.

Length of ER means the change of the regional perimeter through the initialization function with the updated threshold.

$$\varnothing(p) = 4 - 2\left|\{q : qAp \wedge c(q) \leq c(p)\}\right| \tag{6}$$

Euler number of ER means the difference between the number of connected domains and holes. Euler number is the topological feature of two-value images and may be calculated with 2 × 2 templates as shown below.

$$\begin{aligned}
Q_1 &= \begin{Bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0' & 0 & 0' & 0 & 1' & 1 & 0 \end{Bmatrix} \\
Q_2 &= \begin{Bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1' & 1 & 1' & 0 & 1' & 0 & 1 \end{Bmatrix} \\
Q_3 &= \begin{Bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0' & 0 & 1 \end{Bmatrix}
\end{aligned} \tag{7}$$

Q_1, Q_2 and Q_3 represent exterior angle point, interior angle point and intersection. Euler number of ER is calculated by the following formula:

$$\phi(p) = \frac{1}{4}\left(\Delta N_{Q_1} - \Delta N_{Q_2} + 2\Delta N_{Q_3}\right) \tag{8}$$

$\Delta N_{Q_1}, \Delta N_{Q_2}, \Delta N_{Q_3}$ Represent the change of number of points of ER which are defined as above.

The number of intersection of ER means the number of jump of the pixels belonged to r or not on the horizontal scan line in ER, which is shown in Fig. 1.

2.3. High-level HOG features

HOG is a very effective description of text regions because of the significant edge characteristics of text characters. The HOG features can be obtained in each candidate text regions. In view of facilitating the calculation, the candidate regions can be normalized as 16 × 16 and divided into blocks of 4 × 4 and then the gradient information of each block can be obtained as its feature by histogram statistics.

2.4. Normalization and principal component analysis of candidate text regions

As mentioned earlier, six kinds of different feature have been extracted which will be merged into a feature vector x_i (i = 1,2,3,4,5,6). The normalized feature vector is shown below by calculating mean (μ_i) and variance (σ_i) of each kind of feature vector.

$$x_i^* = \frac{x_i - \mu_i}{\sigma_i} \tag{9}$$

After normalization, the mean and variance of the updated feature vector is unified into 0 and 1. Considering the fact that the dimensions of features are different from each other especially the high dimensions of ER, which will be difficult in feature fusion and classification with SVM, principal component analysis will be used to reduce the high-dimension of the feature vector.

3. Support vector machine

3.1. Support vector machine (SVM)

SVM is known as, one statistical learning theory (SLT) [4] which is widely applied in fields of classification, function fitting [5–7], pattern recognition and so on [8–13]. Therefore a lot of researches of SVM have been conducted in recent years [10,14–18].

SVM is a method based on SLT, which is different from the traditional learning methods using the empirical risk minimization criterion that minimizes the error on the training data but shows poor generalization ability. Vapnik proposed criterion of structural risk minimization [19,20], which can improved the generalization ability of classifier by minimizing the upper bound of risk. SVM classifies the samples into different types through optimal hyperplane constructed in samples space which ensures samples of different type can be classified and the distance between the hyperplane and the different types of samples achieves maximum [21,22]. The basic idea of SVM is shown in Fig. 2.

Solid and hollow points represent two kinds of samples, H is the classification line and H_1 and H_2 are the closest parallel line to H in different sample space consists of samples of different type. Distance of H_1 and H_2 is regarded as the margin of classification, the optimal classification line is the line which can classify the samples and ensure the maximum of the margin. Red points on line H_1 and H_2 are called support vectors (SV). Classification methods based on neural networks get the split plane that is often very near to the training samples, which shows the disadvantage of neural networks in small sample space classification compared with SVM. $(x_1,y_1),\cdots,(x_i,y_i), x_i \in R^n, y_i \in \{-1,+1\}$ are the sample points in Fig. 2, and the optimal hyperplane was shown below:

$$w \cdot x + b = 0 \tag{10}$$

"·" is the calculation of inner product, split plane is constructed as below:

$$\begin{cases} w \cdot x_i + b \geq 1, y_i = 1 \\ w \cdot x_i + b \leq -1, y_i = -1 \end{cases} \tag{11}$$

so that we can know:

$$y_i[w \cdot x_i + b] \geq 1 \tag{12}$$

Then the minimum distance from the training samples to the given split plane will be calculated:

$$p(w,b) = \min_{\{x_i | y_i = 1\}} \frac{w \cdot x_i + b}{|w|} - \max_{\{x_i | y_i = -1\}} \frac{w \cdot x_i + b}{|w|} = \frac{2}{|w|} \tag{13}$$

To solve the optimal split plane means the fitness of formula 9 and the maximum of the minimum distance $\left(\frac{1}{2}|w|^2\right)$ which can be described as below:

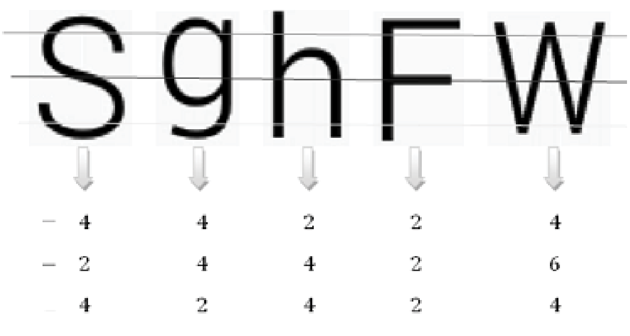

Fig. 1. Number of intersections of ER.

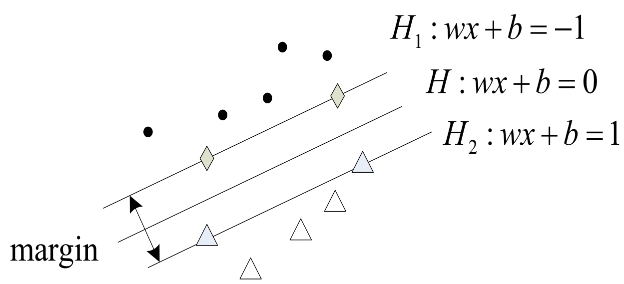

Fig. 2. Optimal classification line of linear separable case.

$$\begin{cases} \min_{w,b} \frac{1}{2}|w|^2 \\ y_i\left(w \cdot x_i + b\right) \geq 1 \end{cases} \quad (14)$$

The input space can be mapped to another feature space of higher dimensions through some pre-selected nonlinear mapping when the training samples are not linearly separable so that the optimal split plane will be achieved in classifying samples of higher dimensions. The classification functions of SVM are the linear combination of nonlinear functions with SV as parameters in the samples data set. Since the classification functions relate to the number of SV but not the spatial dimensions, the changes of spatial dimensions and computation will be considerable when different mapping is applied. SVM solves the problem through some defined kernel functions. The inner product operation in SVM is defined as follows:

$$K\left(x_i, x_j\right) = k\left(x_i\right) \cdot k\left(x_j\right) = \sum a_k \psi_k\left(x_i\right)\psi_k\left(x_j\right) \quad (15)$$

$K(x_i, x_j)$ is the kernel function representing the features of mapping. Computation of $k(x)$ can be avoid at the case of certainty of $K(x_i, x_j)$ so that the computation does not relate to the spatial dimensions.

SVM can also treat multi-value classification through combination [23,24]. Common way of SVM to solve multi-value classifications is the following:

3.1.1. One against the rest method

Classifiers of SVM with number n are constructed and each classifier of them can divide the data into two categories. The classifiers will be trained by the test data and the discriminant values of each classification function will be computed out. The data is regarded as the category that the discriminant value is bigger than the others, which means the categories of test data will be determined by the discriminant value. It is possible that some data can be divided into several categories with the method and data have to be mapped to spatial of higher dimensions with cost of efficiency.

3.1.2. One against one method

Classifiers of SVM with number $\frac{n(n-1)}{2}$ are constructed to achieve the two-class classification in the data set of n-class. Every two classifiers compete with each other and the loser will be rejected. The final winner is the classifier which determines the category of data. There may be data which cannot be divided into any category with the method.

3.1.3. Method of M-ary

Classifiers with number of $\log_2 n$ are constructed through method of M-ary proposed by Sebald [35]. Categories are combined into several new ones so that classification of multi-class will be translated into that of two-class with the decrease of training and testing and increase of efficiency. For example, test data belongs to four categories: {1, 2, 3, 4}, $\log_2 4 = 2$ classifiers will be constructed according to the method. Data of category 1 and 3 will be marked as positive while that of category 2 and 4 as negative for the first classifier; Data of category 1 and 2 will be marked as positive while that of category 3 and 4 as negative for the second classifier; finally test data will be classified with the two classifiers. If the result of a data on the two classifiers is 1 and –1, according to the analysis above the data belongs to not only category of 1 and 3 but also category of 3 and 4, thus the data should be classified into category 3.

3.2. Method of SVM based on multi-kernel functions

In this paper, totally six features of text regions are extracted and each feature can be regarded as the description from aspect of the characteristics of text regions. Considering the limitations of features, Zheng et al. [2] proved future fusion is more effective than single feature. Connecting the features in series forming a feature vector will lead to the losses of the features and the dimensions of features so the method of SVM based on multi-kernel functions [26–28] is adopted in this paper. We use different kernel functions [29–31] for different features and choose the best combination of kernel functions by training the weight of them. Method of SVM based on multi-kernel functions can be described as below:

$$f(x) = \mathrm{sgn}\left(\sum_{i=1}^{N} \alpha_i y_i \sum_{k=1}^{K} \beta_k K_k\left(x_i, x\right) + b\right) \quad (16)$$

$K_k(x_i, x)$ is the kernel function of number K, β_k is the weight of the kernel function, the objective function of the classifier is shown as below:

$$\frac{1}{2}\sum_{k=1}^{K} |w|^2 + C\sum_{i=1}^{N} \varepsilon_i \quad (17)$$

Since every kernel function has its own advantages and disadvantages, each kernel function shows different performance and characteristic so that the performances of SVM are different. Common kernel functions marked as K_1, K_2, such as radial kernel function, linear kernel function, sigmod kernel function, and polynomial kernel function and so on are the usual ones applied in SVM methods based on multi-kernel functions. Methods based on different kernel functions shows the differences of generalization ability [32,33]. For example, radial kernel function shows locality, that is, the points close to each other have influence on the kernel function while those far away from the training data do not. Because of the differences of Riemann metric, the distance measurement and angle measurement in different kernel functions, SVM based on different kernel functions shows different study and generalization ability. Research [34] shows the sample data in the input space can be regarded as the physical particles acting on the whole data set, thus the kernel functions can be chosen to classify the data more effectively according to the measurement of them.

By testing the kernel functions on the text detection with features mentioned above, we finally have chosen radial kernel function for geometric features, sigmod kernel function for local region characters and polynomial kernel function for HOG features, which ensure the locality and good generalization ability. Weights of the kernel functions for the six features are shown in the Table 1 as below.

4. Results and analysis

In this experiment, we picked up one thousand images as the training samples from our own database, got ten thousand images of text character regions as the positive samples by clipping and normalized them to size of 45 × 45. In this same way, we got eight thousand images of non-character regions as the negative samples, which are shown in Fig. 3.

After being trained on the database built above, the algorithm is tested on database of ICDAR2003 and MSRA-TD500. The results are shown in Table 2:

Another compared experiment was made with algorithms of SVM based on single-kernel function and multi-kernel function (results shown in Fig. 4). Algorithm of SVM based on multi-kernel function shows better accuracy with the increase of iterations because kernel functions represent a mapping from low dimensional space to high dimensional space and multi-kernel functions are even in data learning but better in generalization ability compared with single-kernel functions. It should be noted that defined features based on experience may contains some implicit relevance and the kernel functions may have similarities [31,35–43]. In view of the above-mentioned facts, the overall performance of SVM based on multi-kernel functions changes little, which shows the limitations of the features based on experience.

Table 1
Weights of kernel functions

Number of features	Features' id	Weights of kernel functions
1	Aspect ratio of characters	0.1597
2	Uniformity of stroke width	0.1903
3	Length of ER	0.1237
4	Euler number of ER	0.1460
5	Intersection number	0.1355
6	HOG features	0.2448

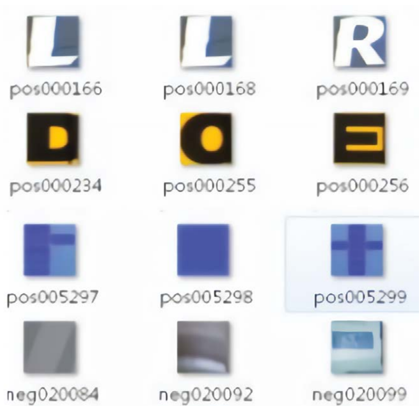

Fig. 3. Database of positive and negative samples.

Table 2
Test results on different database

Database	Time(s)	Recall	Precision	*F*
ICDAR2003	0.083	67.5	77.1	65.3
MSRA-TD500	0.080	69.5	80.4	67.7

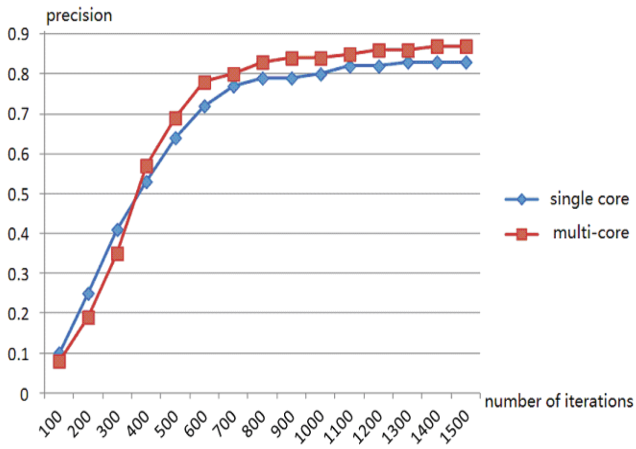

Fig. 4. Compared test between single-kernel SVM and multi-kernel SVM.

5. Conclusion

In this paper, a text detection algorithm based on SVM with multi-kernel functions SVM is proposed. Compared with the traditional text detection algorithms based on single or a few features, the proposed algorithm makes full use of various geometric and high-level features of text regions and fuses the features together into one feature vector. Different features correspond to these different kernel functions. By using multi-kernel functions SVM avoids the loss of difference of the importance of features.

Finally, the proposed algorithm shows satisfied classifications through multi-kernel functions. The experiment results have shown the good performance in both speed and accuracy. Although the limitations of artificial selection of kernel functions are inevitable and the performance of SVM will be affected by kernel functions, the proposed algorithm of SVM still performs well in training and studying, especially in small-sample space. But in large-sample space, algorithms based on machine learning represent the development trend of the future.

Acknowledgements

This paper is funded by the Youth Project of the National Natural Science Foundation of China under Grant 61702355, the Anhui Provincial Natural Science Foundation under Grant 1908085QF283, the Major Projects of Teaching Research in Colleges and Universities of Anhui Province under Grant 2016jyxm1026 and the Key Natural Science Project of Anhui

Provincial Education Department under GrantKJ2018A0448 and KJ2019A1001.

References

[1] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn., 20 (1995) 273–297.

[2] Q. Zheng, K. Chen, Y. Zhou, C.C. Gu, H.B. Guan, Text localization and recognition in complex scenes using local features, Lect. Notes Comput. Sci., 6494 (2011) 121–132.

[3] H. Ouyang, Z. Liu, L. Wang, W. Peng, H. Deng, M.A. Ashraf, Fungicidal activity and bamboo preservation of *Pinus elliottii* needles extracts, Wood Res., 63 (2018) 533–546.

[4] M. Abbasi, U. Rafique, G. Murtaza, M.A. Ashraf, Synthesis, characterisation and photocatalytic performance of ZnS coupled Ag_2S nanoparticles: a remediation model for environmental pollutants, Arabian J. Chem., 11 (2018) 827–837.

[5] B. Epshtein, E. Ofek, Y. Wexler, Detecting Text in Natural Scenes with Stroke Width Transform, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010.

[6] V. Cherkassky, F. Mulier, Learning from Data: Concepts, Theory and Methods, John Wiley & Sons, NY, 1997.

[7] K.S. Rawat, R. Kumar, S.K. Singh, Topographical distribution of cobalt in different agro-climatic zones of Jharkhand state, India, Geol. Ecol. Landscapes, 3 (2019) 14–21.

[8] V. Vapnik, S.E. Golowich, A. Smola, Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing, M.C. Mozer, M.I. Jordan, T. Petsche, Eds., Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, 1997, pp. 281–287.

[9] A. Amid, N.A. Samah, Proteomics as tools for biomarkers discovery of adulteration in slaughtering procedures, Sci. Heritage J., 3 (2019) 11–16.

[10] K.-R. Müller, A.J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, V. Vapnik, Predicting Time Series with Support Vector Machines, ICANN 1997: Artificial Neural Networks — ICANN'97, International Conference on Artificial Neural Networks, Springer Lecture Notes in Computer Science, 1997, pp. 999–1004.

[11] F. Qiao, Research on design principles of visual identity in campus environment, Sci. Heritage J., 2 (2018) 1–3.

[12] H.D. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, V. Vapnik, Support Vector Regression Machines, M.C. Mozer, M.I. Jordan, T. Petsche, Eds., Advances in Neural Information Processing Systems, Morgan Kaufmann, San Mateo, 1997, pp. 155–161.

[13] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, T. Poggio, Pedestrian Detection using Wavelet Templates, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, Puerto Rico, 1997.

[14] A. Amamra, K. Khanchoul, Water quality of the Kebir watershed, northeast of Algeria, J. CleanWas, 3 (2019) 28–32.

[15] M.A. Hearst, B. Scholkopf, S. Dumais, Trends and controversies-support vector machines, IEEE Intell. Syst., 13 (1998) 18–28.

[16] E. Osuna, R. Freund, F. Girosi, Training Support Vector Machines: An Application to Face Detection, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, Puerto Rico, 1997.

[17] T.D.T. Oyedotun, L. Johnson-Bhola, Beach litter and grading of the coastal landscape for tourism development in sections of Guyana's coast, J. CleanWAS, 3 (2019) 1–9.

[18] C.Y. Lu, P.F. Yan, C.S. Zhang, J. Zhou, Face Recognition using Support Vector Machine, Proceedings of ICNNB'98, Beijing, 1998, pp. 652–655.

[19] Y. Rajendran, R. Mohsin, Emission due to motor gasoline fuel in reciprocating lycoming O-320 engine in comparison to aviation gasoline fuel, Environ. Ecosyst. Sci., 2 (2018) 20–24.

[20] V.D. Malsburg, Christoph, V. Seelen, Werner, C. Vorbrüggen, Jan, Sendhoff, Bernhard, Artificial Neural Networks - ICANN 96, 1996 International Conference on Artificial Neural Networks, Bochum, Germany, 1996, pp. 251–256.

[21] M. Wilson, M.A. Ashraf, Study of fate and transport of emergent contaminants at waste water treatment plant, Environ. Contam. Rev., 1 (2018) 1–12.

[22] M. Brown, H.G. Lewis, S.R. Gunn, Linear spectral mixture models and support vector machines for remote sensing, IEEE Trans. Geosci. Remote Sens., 38 (2000) 2346–2360.

[23] A. Ahmed, A. Nasir, S. Basheer, C. Arslan, S. Anwar, Ground water quality assessment by using geographical information system and water quality index: a case study of Chokera, Faisalabad, Pakistan, Water Conserv. Manage., 3 (2019) 7–19.

[24] K. Bennett, O. Mangasarian, Robust linear programming discrimination of two linearly inseparable sets, Optim. Methods Software, 1 (1992) 23–34.

[25] E. Osuna, R. Freund, F. Girosi, An Improved Training Algorithm for Support Vector Machines, Neural Networks for Signal Processing VII. Proceedings of the 1997 IEEE Signal Processing Society Workshop, IEEE, 1997.

[26] L. Yang, H. Guo, H. Chen, L. He, T. Sun, A bibliometric analysis of desalination research during 1997–2012, Water Conserv. Manage., 2 (2018) 18–23.

[27] K.P. Bennett, A. Demiriz, Semi-supervised Support Vector Machines, Proceedings of the 1998 Conference on Advances in Neural Information Processing Systems II, IEEE, 1998.

[28] X.G. Zhang, Using Class-center Vectors to Build Support Vector Machines, Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop, IEEE, 1999, pp. 3–11.

[29] D. Anguita, S. Ridella, S. Rovetta, Circuital implementation of support vector machines, Electron. Lett., 34 (1998) 1596–1597.

[30] V.N. Vapnik, The Nature of Statistical Learning, Springer, Berlin, 1995.

[31] V.N. Vapnik, Statistical Learning Theory, John Wiley & Sons, New York, 1998.

[32] G. Wahba, Spline Models for Observational Data, CBMS-NSF Regional Conference Series in Applied Mathematics, 1990, p. 59.

[33] B. Boser, A Training Algorithm for Optimal Margin Classifiers, Fifth Annual Workshop on Computational Learning Theory, ACM Press, Pittsburgh, 1992.

[34] C.W. Hsu, C.J. Lin, A comparison of methods for multi class support vector machines, IEEE Trans. Neural Networks, 13 (2002) 415–425.

[35] D.J. Sebald, J.A. Buchlew, Support vector machines and the multiple hypothesis test problem, IEEE Trans. Signal Process., 49 (2001) 2865–2872.

[36] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, The Syndicate of the Press of the University of Cambridge, Cambridge, 2000.

[37] H.Q. Wang, F.C. Sun, Y.N. Cai, N. Chen, L.G. Ding, On multiple kernel learning methods, Acta Autom. Sin., 36 (2010) 1037–1050.

[38] H.Q. Yang, Z.L. Xu, J.P. Ye, I. King, M.R. Lyu, Efficient sparse generalized multiple kernel learning, IEEE Trans. Neural Networks, 22 (2011) 433–446.

[39] C. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discovery, 2 (1998) 121–167.

[40] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines: and Other Kernel-Based Learning Methods, Cambridge University Press, New York, 1999.

[41] J.S. Thierman, L.M. Hallaj, Apparatus and Method for Geometric Measurement: U.S. Patent Application 12/784, 694, 2010-05-21.

[42] A. Sohelf, G.C. Karmakar, S. Dooleyls, Geometric distortion measurement for shape coding: a contemporary review, ACM Comput. Surv., 43 (2011) 29.

[43] N. Kevi, S. Zhou, R. Chellappa, From sample similarity: probabilistic distance measure in reproducing kernel hilbert space, IEEE Trans. Pattern Anal. Mach. Intell., 28 (2006) 917–929.